Sensor placement for early detection in compartmental epidemic models on networks

Pau Batlle Franch

Advised by:
Carlos Fernández-Granda (NYU)
Xavier Giró (UPC)

Degree in Mathematics
Degree in Engineering Physics

May 2020

Abstract

This project presents a general framework of node monitorization for any compartmental epidemic model that operates on networks. This can be used to select which nodes in a network should be monitored in order to detect an outbreak following a specified model as fast as possible, make predictions about the current state of the network given the observations of the sensed nodes and decide which nodes should we test immediately after detecting the existence of an epidemic to maximize information gain. In this work I formulate the problems analytically using Markov Chains and propose Monte Carlo sampling algorithms to solve the problems in larger graphs. Finally I illustrate the capabilities of the framework in experiments in two real networks that exemplify two possible uses of the framework in real epidemic scenarios.

Keywords: Epidemiology, Markov Chains, Compartmental models, Submodularity
AMS Code: 60J10

Acknowledgements

I would like to sincerely thank Prof. Carlos Fernández-Granda and Prof. Joan Bruna first for giving me the opportunity to join their groups at NYU from September 2019 to May 2020 in what has been an incredible personal and intellectual experience. I also thank them, together with Prof. Víctor Preciado (UPenn), for providing constant guidance and mentoring during the project with weekly meetings and extended discussions, even remotely during the last couple of months.

My work at NYU has been co-funded by NYU and a CFIS-Cellex Mobility Grant, and I appreciate the effort that both have put into making the possibility of living in New York to pursue this work at NYU viable economically but also logistically.

I would also like to thank some of my closest colleagues at CFIS, Joan, Adam, Marta, Melcior and Pol, for insightful discussions regarding subproblems that have appeared throughout this work, and obviously because the good moments while sharing our adventures travelling abroad to finish our undergraduate degrees.

Finally, I feel very fortunate because my family has provided me with plenty of support both while I was in the United States and lately finishing my work remotely from home, and I sincerely thank them for that.

Figueres, May 2020

Contents

Chapter 1

Introduction

1.1 Project structure

In this project I present my work at NYU during my academic stay from September 2019 to May 2020. The whole body of the project is presented as the scientific article I wrote under the supervision of Carlos Fernández-Granda (NYU), Joan Bruna (NYU) and Victor M. Preciado (UPenn), which is included below this introduction and will appear as a preprint soon after the due date of this work. In this section I provide a small higher level overview to the studied field to contextualize the article.

1.2 Introduction to epidemiology on networks

Mathematical modelling aims to simplify real systems while capturing fundamental properties that can later be used to take decisions, predict outcomes, or, more generally, increase our knowledge about the system. This work explores a tool that can be used to increase the range of questions models of infectious processes can answer.

The most obvious example of infectious processes are infectious diseases, which are very relevant today as the spread of the virus SARS-CoV-2 is causing severe disruptions in economy, public health and society. Mathematical modelling has guided experts into taking measures in this and previous health emergencies, such as Ebola and H1N1 influenza, and also to help eradicate diseases such as Guinea worm and polio.

Various modelling approaches have been used in the field of epidemiology. Ranging from compartmental models, in which population is assumed to be well-mixed, to fully individual agent-based models in which the behaviour of every agent is simulated, different models aim to describe different aspects of the disease evolution. This huge set of models already available in the literature motivate the creation of model-agnostic tools that could rapidly be used out of the box once domain experts have decided which particular model is best used to describe a particular epidemic outbreak. In this work, I present a model-agnostic tool for a subset of models: Markovian models that act on networks. This means that a graph $G = (V, E)$ is assumed to be given, and then the infectious process affects nodes in the network

and travels along its edges. This spread can be stochastic, and the main objective explored in the article is to detect the presence of such processes as fast as possible by monitoring nodes in the network, and in the article I explore which nodes to test and what information can this test give us.

Infectious processes in networks are not limited to infectious diseases. Some other situations in which this framework would apply are the spreading of rumours or fake news in social networks (both online and offline) and the spreading of a computer virus in a network of computers with data transfer between them. For simplicity, most of the notation used throughout will refer to the infectious disease case (i.e nodes will be healthy infected, ...) as in the setup of infectious diseases, but it should be understood that all of it applies to the other settings, in which for example an infectious node could be someone who has heard of a false rumour and now is capable of spreading it to its nearby contacts.

SENSOR PLACEMENT FOR EARLY DETECTION IN COMPARTMENTAL EPIDEMIC MODELS ON NETWORKS

Pau Batlle
Courant Institute of Mathematical Sciences
Center for Data Science
New York University

Carlos Fernandez-Granda
Courant Institute of Mathematical Sciences
Center for Data Science
New York University

Joan Bruna
Courant Institute of Mathematical Sciences
Center for Data Science
New York University

Victor M. Preciado
Department of Electrical and Systems Engineering
Applied Mathematics & Computational Science
University of Pennsylvania

ABSTRACT

We present a general framework of node monitorization for any compartmental epidemic model that operates on networks, which can be used to select which nodes in a network should be monitored in order to detect an outbreak following a specified model as fast as possible, make predictions about the current state of the network given the observations of the sensed nodes and decide which nodes should we test immediately after detecting the existence of an epidemic to maximize information gain. We formulate the problems analytically using Markov Chains and propose Monte Carlo sampling algorithms to solve the problems in larger graphs. We illustrate the capabilities of the framework in experiments in two real networks that exemplify two possible uses of the framework in real epidemic scenarios.

1 Introduction

Epidemic modelling has been the driving force in decision making in order to combat different infectious diseases such as Ebola, H1N1 influenza and most recently Covid-19, and has also been a key factor to help eradicate diseases such as Guinea worm and Polio. A huge amount of models exist with different sets of assumptions and attempting to answer different questions. Since the diseases are spread via close contact between individuals only, networks can be used to effectively encode the information of which individuals can directly infect which other ones, and the study of virus propagation in networks has generated a lot of interest.

With the large amount of available epidemic models, most of which can be easily adapted to networks, the creation of a common framework that does not assume a model beforehand and can answer a set of questions once the model is specified is interesting for a variety of reasons. First of all, it can be used together with any existing modelling approach, giving new insights that might not be obvious from the model alone at virtually no extra modelling cost. In that aspect, the presence of a domain expert that can carefully chose the best model to study the particular situation is key. Secondly, it can also be a useful tool to compare between possible models (or sets of parameters for a single model) that aim to describe reality, as the predictions of the framework can then be compared to real data.

While significant effort has been put in the study of specific questions for different models in networks, such as phase transitions [1] or epidemic thresholds [2], the kind of higher-level questions that a common framework for all compartmental models could aim to solve has yet to be explored. As the evolution studied is inherently Markovian, Markov Chains can been used as powerful tools that can help us solving some of the questions that motivate the creation of the model [3]. Here, we propose a way to solve different problems related to the early stages of an epidemic with the use of Markov Chains. In our setup, we are able to test individuals to detect the existence of an epidemic, whether this disease is new or we want to monitor a new wave of a previous known disease, as it could be the case for Covid-19 as containment measures start to be lifted. Broadly speaking, the most important questions explored in this article are (i)

which nodes should we monitor, (ii) what information would a detection in the monitored nodes give us, and (iii) what nodes should we test next after an outbreak detection.

The main contributions of this work are the following: we present a framework with flexibility to model all kind of compartmental models in networks and we formalize early detection problems in Markov Chain language. Then, we prove a submodularity result for a general coverage-type set function in Markov Chains, and then use it to scale our method to larger graphs, together with Monte Carlo like algorithms.

This article is organized as follows: In section 2, the background of previous literature is presented. In section 3, the main setup for the framework and the questions that we aim to solve is explained. These questions are then rigorously formalized and solved analytically in section 4. Section 5 provides insight into how to solve the problems in practice, as the analytical solution is only usable for very small graphs. Finally, section 6 presents experiments in a toy small network and two real networks in different situations to illustrate the capabilities of the framework.

2 Related Work

General work on stochastic compartmental model on networks include [4], [5], [6]. [7] provides a survey of problems involving spreading processes in networks. More recently, [8] and [9] provide inference procedures with similar goals as the one presented in this paper.

[3] studies the spread of computer viruses in network using Markov Chains in a similar way that we use them in this work, albeit its focus is more on continuous time Markov Chain and the mean field approximations that can be derived from them. [10] uses a similar sensor placement framework as in this work to detect outbreaks such as news spreading or water contaminants with high generality. Here, our first problem is a similar problem but with a different objective function more suited to epidemic spread. However, since our function has similar properties, their methods to scale up greedy-like techniques can apply in this case as well.

Deciding which nodes to monitor is solving a group closeness like-problem, as it is trying to find central nodes in the sense of early epidemic detection. [11] provides an introduction to the different types of flow processes (which include epidemic processes) and how some of them define a closeness measure for nodes. Some work has been done for other types of stochastic flow processes. For example, [12] proposes an algorithm to find nodes that can be reached quickly in random walks over a graph, a problem that presents conceptual similarities to our first problem.

3 Setup: Epidemic, early detection and tracing problems

3.1 Markovian Epidemic Models

All of the questions that we aim to answer given a model correspond to the early stages of an epidemic. In this section we define which kind of models are compatible with the framework and informally state these questions, which we will revisit in the following section using Markov Chains to formalize them and obtain analytical solutions and more precise statements. We consider a given network $G = (V, E)$ with nodes numbered from 1 to n, and a compartmental infection model satisfying the following properties:

1. The model operates in a network in discrete time-steps.

2. At every time-step, each of the nodes is in one of s possible states.

3. The state of the network (i.e of all its nodes) at time $t + 1$ is a stochastic function of the state of the network at time t only.

All classical, compartmental models adapted to networks and discrete-time satisfy these conditions. The simplest, canonical example is given by the classical SIR model of 3 compartments: Susceptible, Infected and Removed [13]. At each time-step, infected nodes may infect healthy network neighbors with probability β and become removed (no longer infectious) with probability γ, as seen schematically in Figure 1.

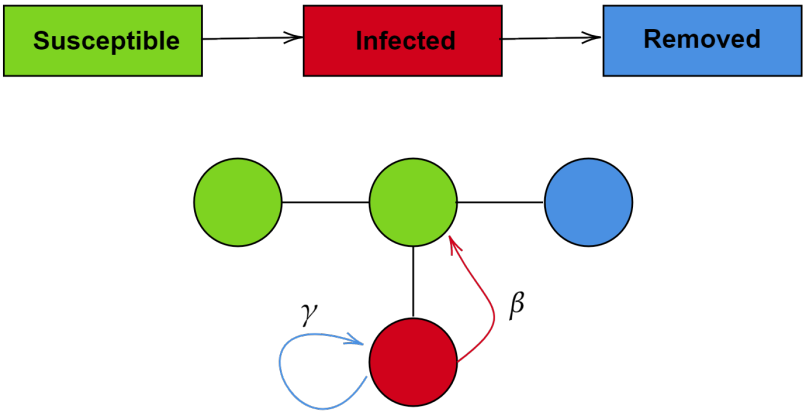

Figure 1: SIR model adaptation to networks and discrete-time. The susceptible, infected and removed nodes are green, red or blue respectively and the arrow colors indicate the possible next state of a node, achieved with probability equal to the number above the arrow

Other examples of models satisfying the hypothesis include the SI (similar to SIR but infected nodes stay infected), SIS (infected nodes become healthy again with probability γ, but they may get re-infected), the Reed-Frost model (the infected individual may spread the disease during a fixed number of timesteps before getting cured) and any compartmental model adapted to network and discrete time [14].

3.2 Sensor Placement and Epidemic Inference Tasks

We assume that we are able to place sensors, each of them letting us monitor the health status of a node in the network. Therefore we can monitor the health of k nodes in the network as time progresses if we have k sensors. The first question (Q1) is precisely related to the placement of these sensors, which can be solved "a priori", before the outbreak. The following questions (Q2-Q5) are "a posteriori" questions, implying that the some of the sensors have already detected the infection, and they relate to the information that we can derive with that observation.

Q1A (Sensor placement with sensor budget): *If we are given k sensors, where should we place sensors in a network to be able to detect an outbreak fast with the highest probability possible? What is the probability of having detected the infection fast enough?*

Q1B (Sensor placement with time budget): *If we are given an objective time in which we should detect the outbreak with a fixed probability, what is the minimum amount of sensors that can achieve that objective? Where should we place them?*

The following questions assume that we have a (not necessarily optimal) solution for Q1, and that we have made a sensor placement. We suppose that after some time of placing the sensors, one or more than one of the sensors signals the infection of their nodes. What can we tell about the infection of the whole at this stage, conditioned to the fact that we have observed that the first time that the infection reached the set of monitored nodes, it reached some specific nodes? Q2 and Q3 ask what information do we have about the past, mostly about where and when the infection outbreak began.

Q2 (Patient zero detection): *Given our observation of the sensors, what is the probability of each initial status (for example each node being patient zero)?*

Q3 (Outbreak time estimation): *Given our observation of the sensors, how much time has happened since the disease started?*

At the time of detection, we know the status of the monitored nodes only, and we want to understand how much information we have about the current status of the rest of the network with the known information so far:

Q4 (Current status assessment): *Given our observation of the sensors, what is the probability of each possible state of the network? In particular, what is the probability that each individual node is infected?*

Finally, in these scenarios, we need to take action to gather more information as soon as possible. Q5 asks about the optimal strategy for near future action. Suppose that after detecting the outbreak at $T = t$, we are given a budget of tests to perform, Q5 asks about the testing strategy that maximizes the information about the current state of the network.

Q5 (Optimal testing after detection): *Given the results of all the previous questions and a budget of tests, which nodes should we test next?*

The following section formalizes these tasks in terms of inferential queries in Markov Chains and provides an analytical solution usable for small graphs only.

4 Theoretical formulation

In this section, we begin by transforming the epidemic models into Markov Chains, and them the five questions from the previous section are formalized and solved analytically.

4.1 Definition of the Markov Chain

Under the three conditions from the previous sections, the epidemic model can be formulated as a Markov Chain of state space $\mathcal{S}$ with $|\mathcal{S}| = s^n$ and transition matrix according to model specifics. The Markov Chain codifies all n nodes in the network, which can each be in the s different compartments. Additionally, a distribution over initial states D needs to be given over the possible initial states of the outbreak. A reasonable choice for the SIR and similar compartmental models with an infected state is that at the beginning of the infection, all nodes are healthy and one is infected, and D becomes a distribution over the n nodes of them being patient zero. Henceforth all probabilities are assumed to be conditioned on X_0, the initial state of the Markov chain, coming from the distribution D.

4.2 Sensor placement optimization (Q1)

Out of the s possible states in which a node might be in the chosen model, we define beforehand which subset of states is detectable by the sensor. Formally, we consider the s states divided in $l \leq s$ non-empty groups, $G_1, ..., G_l$ and the sensor is able to determine the group of the current state only. For the case of $l = s$, this means that the sensor always knows the exact state of the monitored node. A detectable state is then a state which is not in the same group as the state of the population before the infection (susceptible, healthy or similar, depending on the model formalism).

For example, in the SIR model a natural choice is to choose the infected state as the only detectable (whether we choose the removed state as detectable is irrelevant as nodes will always get infected before being removed) and think of the sensor as a method of continued testing over one node that will detect whether the node is infected or not. The objective is to choose the nodes that will be monitored in order to find a detectable state soon after an outbreak happens. In the SIR model example, this means that the infection is detected by the sensors quickly. This notion of detecting the infection is quantified by Markov Chains using hitting times:

Hitting Time: The hitting time of $A \subset \mathcal{S}$, T_A, is the random variable $\min\{n \geq 0 : X_n \in A\}$, where $X_i \in \mathcal{S}$ is the state of the Markov Chain at time i. If the Markov Chain is absorbed in an absorbing state not in A, we set $T_A = \infty$.

Given a subset of nodes $W \subset V$, we also define the detection set of W, D_W, as the subset of Markov Chain states $\mathcal{S}$ consisting of those states in which at least one of the nodes in W is in a detectable state. We therefore have by definition $D_W = \cup_{i \in W} D_{\{i\}}$. This means that given a sensor placement in W, the group of sensors triggers when the Markov Chain reaches one of the states of D_W, although the exact state will still be unknown as some nodes are not monitored. If we fix a set of nodes $W \subset V = \{1, ..., n\}$, then for a Markov Chain in which each node can be in s states, d of which are detectable, we have

$$|D_W| = s^n - s^{n-|W|}(s-d)^{|W|} = s^{n-|W|}(s^{|W|} - (s-d)^{|W|}) \tag{1}$$

This comes from the fact that the states that are not part of D_W are those in which the nodes in W are in one of the $s - d$ non-detectable state, and we don't have any restriction over the rest. By direct counting, there are $s^{n-|W|}(s-d)^{|W|}$ of such states, so subtracting it from the total gives us the desired count for $|D_W|$. We are therefore sending subsets of V to subsets of $\mathcal{S}$ in a way in which the cardinality of the output set only depends on the cardinality of the input set. Since we are only interested of the dynamics before the sensors trigger, we may consider all the states in the detection set as absorbing states in the Markov Chain to calculate the corresponding hitting times.

Now, question **Q1A** can be formalized as follows: Given a time horizon τ, we want to place the k sensors in order to maximize the probability of the Markov chain reaching the detection set in less or equal than τ steps since the epidemic outbreak. Specifically, the optimal placement is given by

$$\underset{W \subset V, |W|=k}{\operatorname{argmax}} \mathbb{P}\left(T_{D_W} \leq \tau\right) = \underset{W \subset V, |W|=k}{\operatorname{argmax}} \mathbb{P}\left(X_\tau \in D_W\right) \tag{2}$$

where we understand that D_W acts as an absorbing set, and so if $X_i \in D_W$ for $i < \tau$ then $X_\tau \in D_W$. Similarly, question **Q1B** is formalized as

$$\underset{W \subset V \text{ s.t } \mathbb{P}\left(T_{D_W} \leq \tau\right) \geq 1-\varepsilon}{\operatorname{argmin}} |W| \tag{3}$$

i.e, the smallest set of sensors that certify a probability larger than $1-\varepsilon$ to detect the epidemic in lesser or equal than τ time-steps.

Computing Hitting Times: We can analytically calculate $\mathbb{P}\left(T_{D_W} \leq \tau\right) = \mathbb{P}\left(T_{D_W} \leq \tau | \, T_{D_W} < \infty\right)\mathbb{P}\left(T_{D_W} < \infty\right)$. $\mathbb{P}\left(T_{D_W} < \infty\right)$ refers to the probability of the Markov Process ending in an absorbing state in D_W. Some models, such as SIR, have absorbing states which represent the epidemic "dying" before any detection in the sensors (for example, the first person gets cured before transmitting it to anyone), which might therefore not be part of D_W. This probability of ending in this group of states can be calculated from the fundamental matrix of the Markov Chain. To calculate $\mathbb{P}\left(T_{D_W} \leq \tau | \, T_{D_W} < \infty\right)$ we need to eliminate these other absorbing states in the chain, which we denote S^-. Given a Markov Chain with two sets of absorbing states S^+ and S^- and a transition matrix M, one can construct another matrix M^+ corresponding to the Markovian dynamics of those processes that get absorbed at S^+ (which we can write as $X_\infty \in S^+$), using the following derivation:

$$\begin{aligned}
M^+_{ij} =& \mathbb{P}\left(X_{t+1} = j | X_t = i, X_\infty \in S^+\right) \\
=& \frac{\mathbb{P}\left(X_{t+1} = j | X_t = i\right) \mathbb{P}\left(X_\infty \in S^+ | X_t = i, X_{t+1} = j\right)}{\mathbb{P}\left(X_\infty \in S^+ | X_t = i\right)} \\
=& \frac{\mathbb{P}\left(X_{t+1} = j | X_t = i\right) \mathbb{P}\left(X_\infty \in S^+ | X_0 = j\right)}{\mathbb{P}\left(X_\infty \in S^+ | X_0 = i\right)} \\
=& M_{ij} \frac{\mathbb{P}\left(X_\infty \in S^+ | X_0 = j\right)}{\mathbb{P}\left(X_\infty \in S^+ | X_0 = i\right)}
\end{aligned} \tag{4}$$

Here, the quantities $\{\mathbb{P}\left(X_\infty \in S^+ | X_0 = i\right)\}_i$ can be found from the fundamental matrix and if i is a state in S^-, or more generally if $\mathbb{P}\left(X_\infty \in S^+ | X_0 = i\right) = 0$, state i will never be reached in the new dynamics and we can safely eliminate it from the chain. This is a proper probability distribution as $\sum_j M_{ij}\mathbb{P}\left(X_\infty \in S^+ | X_0 = j\right) = \mathbb{P}\left(X_\infty \in S^+ | X_0 = i\right)$. The initial distribution also needs to be modified similarly, with $\mathbb{P}\left(X_0 = i | X_\infty \in S^+\right) \propto \mathbb{P}\left(X_\infty \in S^+ | X_0 = i\right) \mathbb{P}\left(X_0 = i\right)$.

Once we have a Markov Chain with the only set of absorbing states S^+, the distribution of hitting times to S^+ follows a discrete phase-type distribution (provided that $X_0 \notin S^+$). If we collapse all the states of S^+ into one (by adding the probabilities that reach it), the hitting times are unchanged and the transition matrix of the Markov Chain with N_t transient states (N_t depends on the choice of compartment model and n) takes the form

$$M = \begin{pmatrix} Q & R \\ 0 & 1 \end{pmatrix} \tag{5}$$

Where Q is a $N_t \times N_t$ matrix, R is a $N_t \times 1$ vector that corresponds to the probabilities of absorption from each state, and there are N_t zeros in the last row.

By definition we have $R + Q\vec{1} = \vec{1}$, where $\vec{1}$ is the vector of all ones. In that case,

$$\mathbb{P}(T_{S^+} \leq k) = \mathbb{P}(T_{S^+} > 0)(1 - vQ^k\vec{1}) + \mathbb{P}(T_{S^+} = 0) \, , \tag{6}$$

where $\boldsymbol{v}$ is the $1 \times t$ vector of starting probabilities for states $s \notin S^+$ (all conditioned to $X_\infty \in S^+$) divided by $1 - \mathbb{P}(T_{S^+} = 0) = 1 - \mathbb{P}(X_0 \in S^+)$ so that the sum of the entries is 1. The full equation reads

$$\begin{aligned}
\mathbb{P}(T_{D_W} \leq \tau) &= \mathbb{P}(T_{D_W} < \infty)\mathbb{P}(T_{D_W} \leq \tau | T_{D_W} < \infty) \\
&= \mathbb{P}(T_{D_W} < \infty) \left(\mathbb{P}(T_{D_W} = 0 | T_{D_W} < \infty) + \mathbb{P}(T_{D_W} \geq 1, T_{D_W} \leq \tau | T_{D_W} < \infty)\right) \\
&= \mathbb{P}(T_{D_W} < \infty) \left(\frac{\mathbb{P}(T_{D_W} = 0)}{\mathbb{P}(T_{D_W} < \infty)} + \mathbb{P}(T_{D_W} \geq 1 | T_{D_W} < \infty)\mathbb{P}(T_{D_W} \leq \tau | T_{D_W} > 0, T_{D_W} < \infty)\right) \\
&= \mathbb{P}(T_{D_W} < \infty) \left(\frac{\mathbb{P}(T_{D_W} = 0)}{\mathbb{P}(T_{D_W} < \infty)} + \left(1 - \frac{\mathbb{P}(T_{D_W} = 0)}{\mathbb{P}(T_{D_W} < \infty)}\right)(1 - vQ^\tau\vec{1})\right) \\
&= \mathbb{P}(T_{D_W} = 0) + (\mathbb{P}(T_{D_W} < \infty) - \mathbb{P}(T_{D_W} = 0)) \left(1 - vQ^\tau\vec{1}\right) \, .
\end{aligned} \tag{7}$$

This objective function could thus be analytically computed if it was possible to operate with the matrix Q. This is very infeasible in practice as Q scales roughly as M, which is of size $s^n \times s^n$. Even then there are $\binom{n}{k}$ possible sensor placements, which also makes it infeasible to solve the problem without a proper optimization scheme that avoids computing the objective function for all possible placements. A similar problem will occur in the following questions. To overcome these we rely on Monte Carlo simulations and submodularity function optimization results, which we discuss in the following section.

4.3 A posteriori inference tasks (Q2, Q3, Q4)

From now on the following situation is assumed: We have sensors placed in a subset W of nodes of the network, with $|W| = k$, and at the current time, at least one of them has detected a detectable state for the first time. The situation is represented schematically in Figure 2. The state of the nodes with sensors is known to a certain degree, but the state of nodes outside those is not determined a priori.

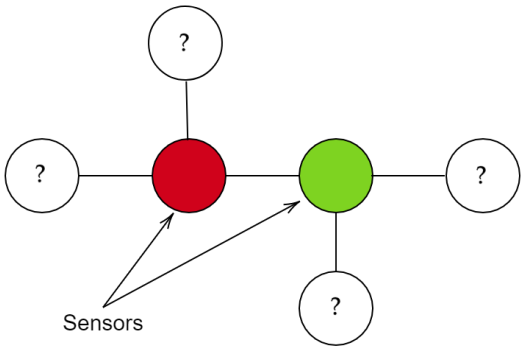

Figure 2: Example situation after sensor detection. Here green or red might mean a specific state or a group of states, depending on the power of the sensors

We define a state of the Markov Chain to be *compatible* with our observation if in that state the sensed nodes are in a state which agrees with the sensors. These include all possibilities for non-sensed nodes but may include some variations in sensed-nodes if the sensors do not perfectly distinguish all states. We let $\mathcal{C} \subset D_W \subset \mathcal{S}$ be the set of compatible states, and O denote our observation of the sensors.

Q2 (Patient zero detection): Q2 asks about the posterior distribution of initial state after our observation of the nodes, given a known prior D from which X_0 comes from. We can apply Bayes' Theorem

$$\mathbb{P}(X_0 = x|O) = \frac{\mathbb{P}(O|X_0 = x)\mathbb{P}(X_0 = x)}{\mathbb{P}(O)} \propto \mathbb{P}(O|X_0 = x)\mathbb{P}(X_0 = x)\,, \tag{8}$$

as $\mathbb{P}(O)$ is just a constant that ensures $\sum_{i\in\mathcal{S}} \mathbb{P}(X_0 = i|O) = 1$. We know $\mathbb{P}(X_0 = x)$ as the initial distribution D is known. For $\mathbb{P}(O|X_0 = x)$, we again consider all states of the detection set of the placed sensors as absorbing, and we need to sum the probabilities of getting absorbed to exactly those states in the detection set which are compatible with our observation. Therefore

$$\mathbb{P}(O|X_0 = x) = \sum_{\alpha\in\mathcal{C}} \mathbb{P}(X_\infty = \alpha|X_0 = x)\,. \tag{9}$$

The probability of ending in a specific absorbing state starting from a specific transient space can be found with the fundamental matrix of the Markov Chain.

Q3 (Outbreak time estimation): Q3 asks about the distribution of time since the epidemic began. We denote $\mathbb{P}(t = k)$ the probability of the infection having started exactly k timesteps ago (this is, the process started at $t = 0$ and was absorbed at D_W at $t = k$), and to calculate it we start similarly as in Q2:

$$\mathbb{P}(t = k|O) = \frac{\mathbb{P}(O|t = k)\mathbb{P}(t = k)}{\mathbb{P}(O)} \propto \mathbb{P}(O \cap (t = k))\,. \tag{10}$$

In the analytical calculation of Q1 we have derived how to calculate the distribution of the hitting time of D_W, so this accounts for the prior distribution $\{\mathbb{P}(t=k)\}_{k\in\mathbb{N}}$. Also similarly as before,

$$\mathbb{P}(O \cap (t=k)) = \sum_{\alpha\in\mathcal{C}} \mathbb{P}((X_\infty = \alpha) \cap (t=k)) \,. \tag{11}$$

$\mathbb{P}(X_\infty = \alpha \cap t = k)$ is the probability that absorption to state α happens exactly at time k. This is $\mathbb{P}(X_k = \alpha) - \mathbb{P}(X_{k-1} = \alpha)$, both conditioned on $X_0 \sim D$ and which can be calculated with powers of the Markov Chain transition matrix.

Q4 (Current status assessment): Q4 asks generally about the probability distributions of the current state over the states in D_W. This is calculated using the same idea as Q2 and Q3. We denote X the actual state

$$\mathbb{P}(X=x|O) = \frac{\mathbb{P}(O|X=x)\mathbb{P}(X=x)}{\mathbb{P}(O)} \propto \mathbb{P}(O|X=x)\mathbb{P}(X=x) = \mathbb{I}(x\in\mathcal{C})\mathbb{P}(X=x) \,. \tag{12}$$

$\mathbb{P}(X=x)$ is the probability of being absorbed at state x, which is known a priori with the fundamental matrix. Therefore, we see that the observation just restricts the probability distribution from D_W to its subset $\mathcal{C}$. We can now solve for the probability of node i being in state s_l by summing over the posterior probabilities of all states in which i is in s_l.

4.4 Optimal testing (Q5)

We aim to perform those tests that give us maximum information about the state of the network. As with the sensors, we may model tests as being able to distinguish between some states and unable to distinguish between others. A test will therefore tell us that a node is part of a subset of the s possible states. Similarly as before, this choice just alters the set of compatible states.

We use the classical concepts of entropy and mutual information to formalize our objective of gaining maximum information. Before doing any additional tests, our distribution of states calculated in Q4 $X|O$ has en entropy of

$$H(X|O) = -\sum_{x\in\mathcal{S}} \mathbb{P}(X=x|O)\log\mathbb{P}(X=x|O) \,. \tag{13}$$

We aim to perform some test T to a fixed number of nodes, so that after observing the result of them the entropy of X has decreased the most on average. The average entropy of X after the test to a subset of nodes W', $T_{W'}$, which may take values $t_1,\dots,t_k$, with $k = d'^{|W'|}$ if the test is able to distinguish between d' groups of states, is:

$$\begin{aligned} H(X|O,T_{W'}) &= -\sum_{i=1}^{k} \mathbb{P}(T_{W'}=t_i) H(X|O,T_{W'}=t_i) \\ &= -\sum_{i=1}^{k} \mathbb{P}(T_{W'}=t_i) \sum_{x\in\mathcal{S}} \mathbb{P}(X=x|O,T_{W'}=t_i)\log\mathbb{P}(X=x|O,T_{W'}=t_i) \,, \end{aligned} \tag{14}$$

and therefore the problem we aim to solve is the maximization of the mutual information

$$\operatorname*{argmax}_{W'\subset V} I(X|O;T) = \operatorname*{argmax}_{W'\subset V} H(X|O) - H(X|O,T_{W'}) = \operatorname*{argmin}_{W'\subset V} H(X|O,T_{W'}) \,. \tag{15}$$

For example, if $W'=W$, and provided that the tests are not more powerful than the sensors, the mutual information would be zero. This is the logical conclusion that testing the already known nodes is the worst thing one could do, as we always have $H(X|O) \geq H(X|O,T_{W'})$. If we perfectly know $P(X=x|O)$, then we are able to evaluate the expression for $H(X|O,T_{W'})$, since if $\mathcal{C}_i \subset \mathcal{C}$ are the compatible states with test output t_i, $\mathbb{P}(T_{W'}=t_i) = \sum_{\alpha\in\mathcal{C}_i}\mathbb{P}(\alpha|O)$ and similarly as before,

$$\mathbb{P}(X=x|O,T_{W'}=t_i) \propto \mathbb{P}(T_{W'}=t_i|O,X=x)\mathbb{P}(X=x|O) = \mathbb{I}(x\in\mathcal{C}_i)\mathbb{P}(X=x|O) \,. \tag{16}$$

We could now theoretically evaluate the mutual information for all tests to obtain the best one. As in the sensor case, the brute force approach tries an exponential number of possibilities, and access to an estimator of $\mathbb{P}(X=x|O)$, which

in turn require an exponential number of roll-outs. Alternatively, one could instead attempt to estimate the marginals of X (i.e, the probability of each node being in each state after the sensor detection). This only requires a number of runs scaling roughly as the number of nodes to maintain estimation accuracy. Recall that when we write $\mathbb{P}(X = x|O)$, x, an state of the Markov Chain, actually encapsulates the states of all nodes $S_1, S_2, \ldots, S_n$, all of which take values in $1, 2, \ldots, s$. Therefore, for the $\{s_i\}$ determined by x we have

$$\mathbb{P}(X = x|O) = \mathbb{P}(S_1 = s_1, S_2 = s_2, \ldots, S_n = s_n|O) \, . \tag{17}$$

Assuming that we have access to the marginals $\{\mathbb{P}(S_i = s_j|O)\}_{i,j}$, what can we say about the information gain? The marginals do not completely determine the full distribution, and so all the values of information gain depend on the copula of the n random variables S_i. For the simplest copula, which is the assumption that the S_i are independent, the distribution factorizes, so $\mathbb{P}(S_1 = s_1, S_2 = s_2, \ldots, S_n = s_n|O) = \prod_{i=1}^{n} \mathbb{P}(S_i = s_i|O)$, and if we assume that the testing process does not destroy this independence this leads to

$$H(X|O, T_{W'}) = H((S_1, S_2, \ldots, S_n)|O, T_{W'}) = \sum_{i=1}^{n} H(S_i|O, T_{W'}) = \sum_{i \in W'} H(S_i|O, T_{W'}) + \sum_{i \notin W'} H(S_i|O) \, , \tag{18}$$

where the last equality comes from the fact that a test on W' does not change our knowledge of other nodes under this independence assumption. Now,

$$\begin{aligned} H(X|O) - H(X|O, T_{W'}) &= \sum_{i=1}^{n} H(S_i|O) - \sum_{i \in W'} H(S_i|O, T_{W'}) - \sum_{i \notin W'} H(S_i|O) = \\ &= \sum_{i \in W'} \left(H(S_i|O) - H(S_i|O, T_{W'}) \right) \\ &= \sum_{i \in W'} \left(H(S_i|O) - H(S_i|O, T_{\{i\}}) \right) . \end{aligned} \tag{19}$$

This means that we can effectively rank the nodes, with a score equal to $H(S_i|O) - H(S_i|O, T_{\{i\}}) \geq 0$ and take in W' the desired number of elements of the highest score as we want to maximize the mutual information. This result is fairly intuitive: For example, in a simplified case of just infected and healthy states, if we estimate that a node is healthy with probability 0.5 and infected with probability 0.5 after the sensor observation and a test could tell whether it is healthy or infected, it is more beneficial to test it rather than someone else with 0.9 probability of being infected as far as information of the current state of the pandemic is concerned. This score can also be computed with our available information about the marginals: by definition

$$H(S_i|O) = -\sum_{j=1}^{n} \mathbb{P}(S_i = s_j|O) \log \mathbb{P}(S_i = s_j|O) \, . \tag{20}$$

$H(S_i|O, T_{\{i\}})$ depends on the capacity to distinguish states. If all test outputs completely determine the state of the node, then $H(S_i|O, T_{\{i\}}) = 0$ with probability one, since the resulting discrete distribution assigns probability one to a state and 0 elsewhere. Similarly to the sensor case, the generalized version is considering the s states divided in l groups, $G_1, ..., G_l$. The test can output l possibilities and the test outputting x, $T_{\{i\}} = x$, means that the node is in a state in group G_x. In this case,

$$\begin{aligned} H(S_i|O, T_{\{i\}}) &= \sum_{j=1}^{l} \mathbb{P}(T_{\{i\}} = j|O) H(S_i|O, T_{\{i\}} = j) \\ &= \sum_{j=1}^{l} \left(\sum_{\alpha \in G_j} \mathbb{P}(S_i = \alpha|O) \right) H(S_i|O, T_{\{i\}} = j) \\ &= -\sum_{j=1}^{l} \left(\sum_{\alpha \in G_j} \mathbb{P}(S_i = \alpha|O) \right) \sum_{k \in G_j} \mathbb{P}(S_i = k|O, T_{\{i\}} = j) \log \mathbb{P}(S_i = k|O, T_{\{i\}} = j) \\ &= -\sum_{j=1}^{l} \left(\sum_{\alpha \in G_j} \mathbb{P}(S_i = \alpha|O) \right) \sum_{k \in G_j} \frac{\mathbb{P}(S_i = k|O)}{\sum_{\alpha \in G_j} \mathbb{P}(S_i = \alpha|O)} \log \left(\frac{\mathbb{P}(S_i = k|O)}{\sum_{\alpha \in G_j} \mathbb{P}(S_i = \alpha|O)} \right) \\ &= -\sum_{j=1}^{l} \sum_{k \in G_j} \mathbb{P}(S_i = k|O) \log \left(\frac{\mathbb{P}(S_i = k|O)}{\sum_{\alpha \in G_j} \mathbb{P}(S_i = \alpha|O)} \right) \, . \end{aligned} \tag{21}$$

This provides a criteria for the best nodes to test next under independence assumptions using only information about the marginal distributions, which are more feasible to estimate than the whole distribution. Similarly as in Q1, the analytical solutions for Q2-Q4 rely on being able to operate with matrices of order s^n, prohibitively large for just a moderate amount of nodes, while Q5 requires estimating the marginals. Next section covers algorithms to solve all of the problems in practise without these unattainable memory requirements.

5 Solving the problem in practice with Monte Carlo sampling

We assume now that we have access to a simulator, this is, a program to obtain samples from the Markov Chain. This requirement is far weaker than being able to store the exponential-growing transition matrix of the Markov Chain, as we can simulate one step just by simulating the random processes that each node undergoes to find the next state. For each run, we simulate the Markov Chain until one of two things happen: We reach a true absorbing state of the Markov Chain (these do not include the added absorbing states related to sensor placement, but they do include for example all healthy or all removed in the SIR model), or all the nodes have already been in a detectable state at least once. We therefore keep track of nodes that have already been in a detectable state in the current run. Algorithm 1 provides an example of the typical structure of a simulator capable of generating N_R runs from a given Markov Chain representing a specific model.

Algorithm 1: Example of simulator

Input: $N_R \in \mathbb{N}$, network G of N nodes, initial distribution D, simulator of one step of the model OneStep
Output: N_R runs of a specific epidemic model on network G

```
Runs ← [] ;
for i = 1, ..., N_R do
    Sample X_0 ~ D ;
    CurrentState ← X_0 ;
    CurrentRun ← [CurrentState];
    detect ← zeros(1,n) ;                                   /* Keep track of detected nodes */
    while not (CurrentState absorbing or ∀i, detect[i] = 1) do
        NextState ← OneStep(CurrentState) ;
        CurrentRun ← CurrentRun ∪ CurrentState ;
        CurrentState ← NextState ;
        for n = 1, ..., N do
            if detect[n] = 0 and n detectable in CurrentState then
                detect[n] ← 1;
    Runs ← Runs ∪ CurrentRun
return S
```

We now revise all problems to explain how to use the simulator to obtain approximate solutions in practice

5.1 Sensor placement optimization (Q1)

We define the optimization objective function over subsets of nodes for a given τ as

$$\begin{aligned} f\colon \mathcal{P}(V) &\to \mathbb{R} \\ W &\to \mathbb{P}\left(T_{D_W} \leq \tau\right) , \end{aligned}$$

so that our objectives become

$$\underset{W\subset V,|W|=k}{\operatorname{argmax}} f(W) \text{ , or } \underset{W\subset V, f(W)\geq 1-\epsilon}{\operatorname{argmin}} |W| \text{ .} \tag{22}$$

5.1.1 Function evaluation

f can be approximated with Monte Carlo samples as follows: Let L be a matrix of shape (N_R, n) such that, for every run $i \in \{1, 2, \ldots, N_R\} = [N_R]$

$$L[i, j] = \min\{n \in \mathbb{N} | \text{Node } j \text{ is detectable at } t = n \text{ in run } i\} , \tag{23}$$

with $L[i, j] = \infty$ if node j is never detectable in run i. The process of creating and populating this matrix can be done inside the simulator at no extra cost (In fact, it can be used instead of the detect binary vector). Given this matrix, an estimator for f, $\hat{f}$, can be calculated as follows:

$$\hat{f}\colon \mathcal{P}(V) \to \mathbb{R}$$
$$W \to \frac{|i \in [N_R] \text{ s.t} \min_{k\in W} L[i,k] <= \tau|}{N_R}\,.$$

As $N_R \to \infty$, we have that $\hat{f} \to f$ uniformly, since

$$\frac{1}{N_R}|i \in [N_R] \text{ s.t} \min_{k\in W} L[i,k] \leq \tau| \to \mathbb{P}(T_{D_W} \leq \tau) \tag{24}$$

because of the law of large numbers and the domain $\mathcal{P}(V)$ is finite. We therefore approximate f by evaluating $\hat{f}$ using all the available Monte Carlo samples.

5.1.2 Optimization of f via submodularity

The combinatorial structure of the problem requires not only a way to rapidly evaluate the objective function but an optimization scheme that avoids evaluating the exponential number of possible sensor placements. In order to do that, we exploit the submodularity properties of f and fundamental results about submodular optimization. If Ω is a finite set, a function $h\colon \mathcal{P}(\Omega) \to \mathbb{R}$ is called submodular if it satisfies one of these three equivalent conditions

1. $\forall X, Y \subseteq \Omega$ with $X \subseteq Y$ and every $x \in \Omega \setminus Y$ we have that $h(X \cup \{x\}) - h(X) \geq h(Y \cup \{x\}) - h(Y)$
2. $\forall S, T \subseteq \Omega$ we have that $h(S) + h(T) \geq h(S \cup T) + h(S \cap T)$
3. $\forall X \subseteq \Omega$ and $x_1, x_2 \in \Omega \backslash X$ such that $x_1 \neq x_2$, $h(X \cup \{x_1\}) + h(X \cup \{x_2\}) \geq h(X \cup \{x_1, x_2\}) + h(X)$

We aim to prove that our defined f is non-negative, monotone (i.e that $f(X) \leq f(Y)$ for $X \subset Y$) and submodular. The non-negativity is trivial from the definition of probability, and monotonicity comes from the fact that for $A \subset B$, $D_A \subset D_B$ and so the event $T_{D_A} \leq \tau$ always implies $T_{D_B} \leq \tau$, so $f(A) \leq f(B)$. We prove in the Appendix that f is sub-modular:

Theorem 1. *f as defined in this section is a submodular function.*

We can now invoke two well-known results in submodular function optimization that will give us guarantees of greedy-like optimization for the two problems mentioned above, using algorithms 2 and 3 respectively. Both of run in polynomial time and only require $\mathcal{O}(n^2)$ evaluations of the objective function.

Theorem 2. *(Nemhauser et al., 1978) If f is monotone submodular and non-negative, the greedy scheme in Algorithm 2 applied to the problem $\operatorname{argmax}_{W\subset V,|W|=k} f(W)$ returns a solution S' for which $f(S') \geq (1 - \frac{1}{e})f(S^*)$ where S^* is the optimal set. [15]*

Algorithm 2: Greedy scheme for Q1A

Input: $k \in \mathbb{N}$, f function over sets of V
Output: $S \subset V$ with $|S| = k$ approximate solution to Q1A
$S \leftarrow \emptyset$;
$i \leftarrow 0$;
while $i \leq k$ **do**
  $S \leftarrow S \cup \underset{v\in V\setminus S}{\operatorname{argmax}} f(S \cup \{v\})$;
  $i \leftarrow i + 1$;
return S

Theorem 3. *(L. Wolsey, 1982) If f is submodular, the greedy scheme in Algorithm 3 applied to the problem $\operatorname{argmin}_{W\subset V, f(W)\geq 1-\epsilon} |W|$ returns a solution S' for which $\frac{|S'|}{|S*|} \leq 1 + \log \frac{f(V) - f(\emptyset)}{f(S') - f(S_{-1})}$ where S* is the optimal set and S_{-1} is the set at the iteration prior to the termination of the algorithm. [16]*

Algorithm 3: Greedy scheme for Q1B

Input: $\varepsilon \in [0,1]$, f function over sets of V
Output: $S \subset V$ with $f(S) \geq 1-\varepsilon$ approximate solution to Q1B
$S \leftarrow \emptyset$;
while $f(S) < 1-\varepsilon$ **do**
  $S \leftarrow S \cup \underset{v \in V \setminus S}{\operatorname{argmax}} \hat{f}(S \cup \{v\})$
return S

It is however our case that we are not optimizing f by using evaluations of f itself, but an approximation $\hat{f}$. A reasonable question is whether $\hat{f}$ has similar properties as f that can guarantee some optimization success. Our next result proves that, at least optimization wise, optimizing with the greedy scheme with samples of $\hat{f}$ is not worse than optimizing with samples of f, as $\hat{f}$ satisfies the same submodularity, non-negativity and monotonicity conditions.

Theorem 4. *The sample approximation of f, $\hat{f}$, is non-negative, monotone and submodular for all values of N_R*

Proof. Non-negativity is true by definition, and monotonoicity comes from the fact that if $A \subset B$, $min_{k\in A} L[i,k] \geq min_{k\in B} L[i,k]$ and so for an equal number or less of runs the minimum over A will be $\leq \tau$ than the minimum over B, so $f(A) \leq f(B)$. For submodularity, we want to see, that for $X \subset Y$ and $x \in V \setminus Y$

$$\hat{f}(X \cup \{x\}) - \hat{f}(X) \geq \hat{f}(Y) - \hat{f}(X \cup \{x\}) \,. \tag{25}$$

The left hand side is $|i \in [N_R] \text{ s.t } \min_{k\in X\cup\{x\}} L[i,k] <= \tau| - |i \in [N_R] \text{ s.t } \min_{k\in X} L[i,k] <= \tau| = |i \in [N_R] \text{ s.t } (\min_{k\in X} L[i,k] > \tau) \cap (L[i,x] \leq \tau)|$. Similarly, the right hand side is $|i \in [N_R] \text{ s.t } (\min_{k\in Y} L[i,k] > \tau) \cap (L[i,x] \leq \tau)|$. As $\min_{k\in Y} L[i,k] > \tau \implies \min_{k\in X} L[i,k] > \tau$, there are at least as many elements in the set of the left hand side than in the set of the right hand side, proving the inequality. □

Taking limits in the submodularity inequality for $\hat{f}$ provides an alternative proof that f is submodular.

5.2 Q2-Q3 implementation

For Q2 and Q3, we are going to use the usual sample estimators for conditioned probability. For both questions, Obs_i refers to the observation our sensors would have observed if the real infection had happened exactly like run i. This can be obtained with the sequence of states of the run, seeing which one is the first in D_W, and may or may not match our true sensor observation O.

This means that given the runs, the approximate solution for Q2 (Patient zero detection) is that the distribution over initial states is:

$$\hat{\mathbb{P}}(X_0 = x | Obs = O) = \frac{|i \text{ s.t } X_0^i = x \cap Obs^i = O|}{|i \text{ s.t } Obs^i = O|} \,. \tag{26}$$

This will not be a good estimator if the set of initial states is exponential with the number of nodes. As mentioned before, however, a reasonable assumption is that D is a distribution over nodes about the probabilities of them becoming patient zero in the infection, so it scales with n. For Q3 (Outbreak time estimation), our empirical distribution of times depends on the how much time it took for the observed runs which end at $Obs^i = O$ to get absorbed in this state:

$$\hat{\mathbb{P}}(t = k | Obs = O) = \frac{|i \text{ s.t } t_{ABS}^i = k \cap Obs^i = O|}{|i \text{ s.t } Obs^i = O|} \,. \tag{27}$$

5.3 Marginal distributions approximation for Q4-Q5

For Q4 (Current status assessment), we do not expect to have sufficient samples to be able to accurately approximate the exact distribution over the exponential number of states, so we directly focus on approximating the marginal distributions of nodes $\{\mathbb{P}(S_k = s_j | O)\}_{k=1:n, j=1:s}$. We approximate this by looking at the network just at the time it produces observation O, and obtaining statistics over node status. S_k^i is the status of node k at the time of detection of run i.

$$\hat{\mathbb{P}}(S_k = s_j | Obs = O) = \frac{|i \text{ s.t } S_k^i = s_j \cap Obs^i = O|}{|i \text{ s.t } Obs^i = O|} \,. \tag{28}$$

Using these approximated marginal distributions, one can directly solve Q5 (Optimal testing after detection) under the independence assumption, as we shown in the previous section how to obtain the mutual information for testing node k as a function of the marginals, and then we test first those nodes with highest mutual informations:

$$-\sum_{j=1}^{n}\mathbb{P}(S_k=s_j|O)\log\mathbb{P}(S_k=s_j|O)+\sum_{j=1}^{l}\sum_{g\in G_j}\mathbb{P}(S_k=g|O)\log\left(\frac{\mathbb{P}(S_k=g|O)}{\sum_{\alpha\in G_j}\mathbb{P}(S_k=\alpha|O)}\right), \tag{29}$$

in which we replace all true probabilities $\mathbb{P}(S_k=s_j|O)$ with our approximates $\hat{\mathbb{P}}(S_k=s_j|O)$

6 Experiments

We present two groups of experiments: the first is a small toy example in which all questions have been solved analytically to illustrate the results that applying the whole framework can have. The second group corresponds to applying the approximate algorithms explained in Section 5 to two real networks in different situations, acting as a reasonable use cases for epidemic outbreak detection. A SIR model with $\beta=0.5$, $\gamma=0.25$ is used in all cases.

6.1 Toy example on a small graph

We use the graph with $n=7$ nodes and $m=7$ edges in Figure 3, and set $\tau=3$. The sensors that maximize $\mathbb{P}\left(T_{D_W}\leq\tau\right)$ are nodes 1 and 5 with a value of 0.850 (circled in the figure)

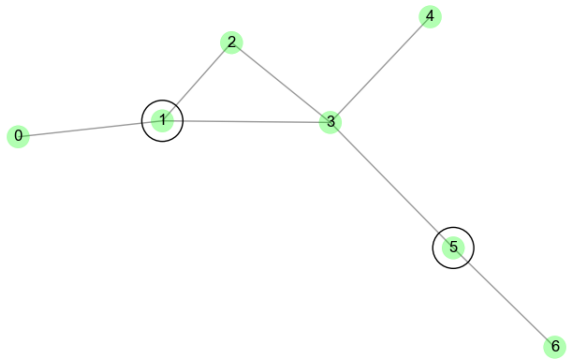

Figure 3: Toy example graph of $n=7$ nodes and $m=7$ edges. For $\tau=3$, nodes 1 and 5 are the optimal set of two sensors

We then run a simulation in which the infection starts from an unknown initial node, and the sensors detect when a node gets infected. The first time the sectors detect the existence of the epidemic, both nodes 1 and 5 test positive for the infection. The status of the rest of the network is initially unknown. We calculate the posterior distribution of patient zero probabilities, which can be seen in figure 4. Note that nodes 0, 1, 5, 6 are given 0 probabilities. Indeed, if 0 or 1 were the initial nodes, the infection would have reached 1 before 5, and so the sensor reading would have not been simultaneous infection of 1 and 5, and similarly for 5 and 6 and node 5.

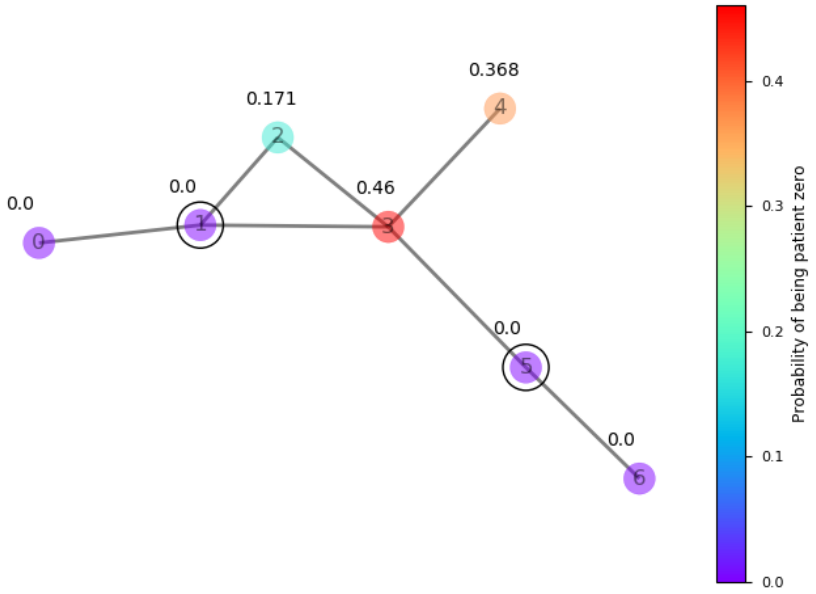

Figure 4: Posterior distribution of patient zero probabilities for the toy example graph given simultaneous observation in nodes 1 and 5

The posterior distribution of times can be seen in Figure 5. The observation of simultaneous infection of two nodes discards the possibility of having observed the infection at $t = 0$

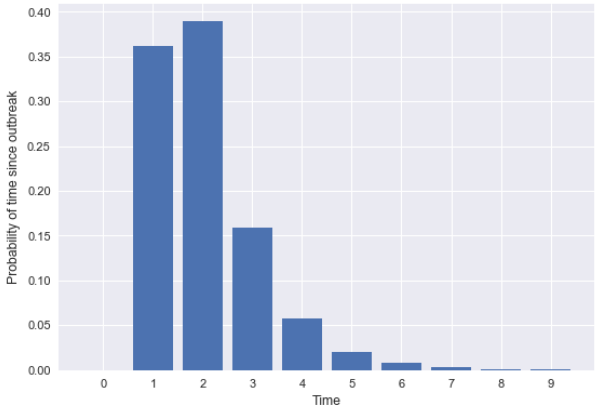

Figure 5: Posterior distribution of time since the infection started for the toy example graph given simultaneous observation in nodes 1 and 5

Finally, we calculate the marginals for each node and their entropies. In Figure 6, node color represents the entropy of their marginal distribution, plotted next to each node. We see for example how node 3 has been infected with probability 1. Indeed, if node 1 and 5 get infected simultaneously, this implies that the infection has travelled through 3, as it appears in the two paths from 1 to 5. There is some probabililty that 3 is not infectious anymore. If we were given tests to test the network immediately, and these tests were able to distinguish between the the statuses, the gain to obtain the maximum amount of information is to test node 4 first.

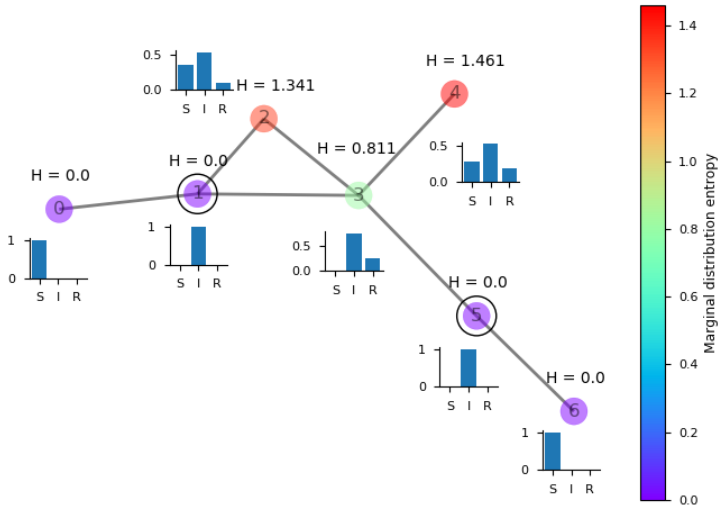

Figure 6: Marginal distribution of each node belonging to each state and entropy of the distributions for the toy example graph given simultaneous observation in nodes 1 and 5

6.2 Real networks

We test the approximate algorithms in two real networks which can act as reasonable cases of interest in epidemic tracing.

6.2.1 US air transportation network

We use the US air transportation network [17], in which we preemptively want to place sensors able to detect if a node is infected. These can be interpreted as a continuous monitorization of passengers in specific airports. This network has $n = 332$ and $m = 2126$, far beyond the range of analytical calculations. A similar analysis as in the toy example is performed, with $\tau = 3$ but in this case $N_R = 10^6$ runs are performed and $k = 10$ sensors are placed according to the greedy scheme in Algorithm 2. The obtained solution can be seen in Figure 7, where the number before each selected airport is the order of choice in the greedy scheme. Their approximate probability of detecting the infection in lesser or equal than 3 steps is 0.93. In comparison, the mean probability of a sample of 10^5 random sensor placement is 0.82, and its maximum is still below the greedy solution at 0.89.

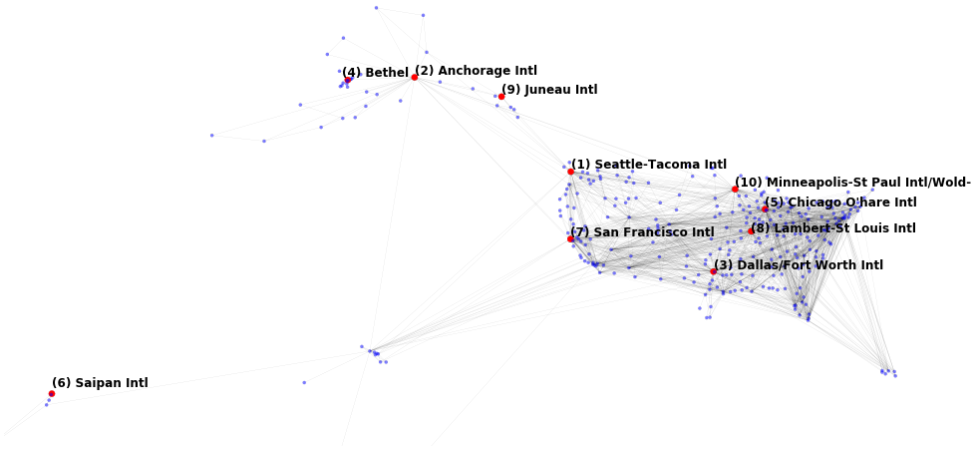

Figure 7: Overview of the US air transportation graph, with 10 sensor placement obtained using Algorithm 2 and a sample approximation of the objective function with 10^6 runs. The number before the airport name indicates the order of placement by the greedy algorithm

Now, similarly as in the previous case, an infection is simulated starting from a theoretically unknown node (in this case, it was Fayetteville in North Carolina, but the algorithm can not use that information) until the sensors are infected. In this case, the observation can be seen in Figure 8, where infected sensed nodes appear in red and healthy sensed nodes appear in green.

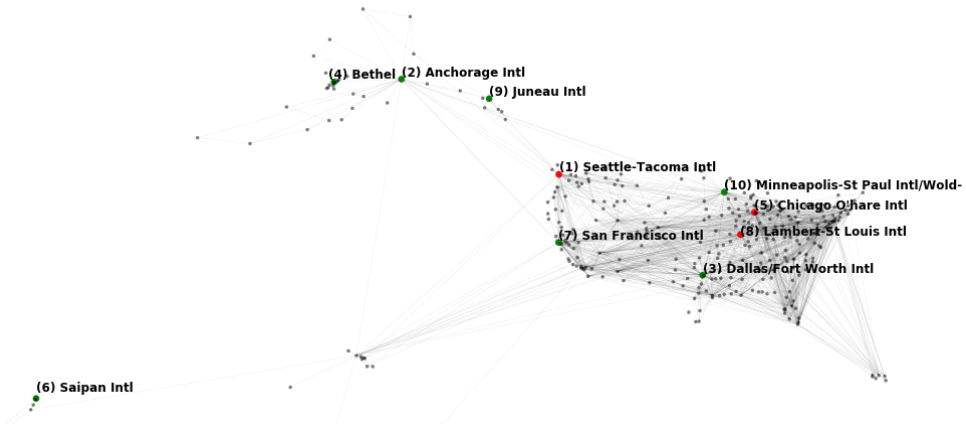

Figure 8: State of the epidemic observed by the sensors in the $k = 10$ nodes selected. Sensors are able to detect if a node is healthy (green) or infected (red)

We perform the approximate inference of the posterior distribution of patient zero probabilities, the results of which can be found in Figure 9, in which node size is proportional to its probability.

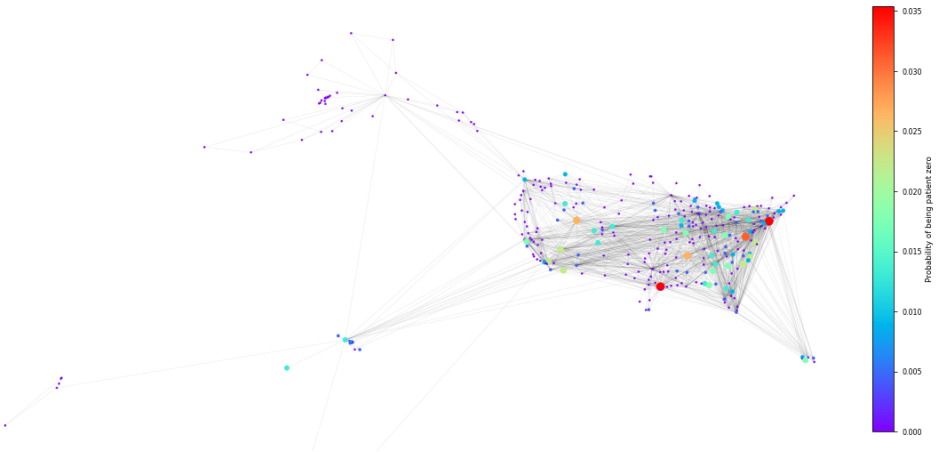

Figure 9: Probability distribution of being patient zero, after the detection in Figure 8 is taken into account. Node size is proportional to probability

The posterior results for time since initial infection can be found in Figure 10. The probability of $t > 2$ is small and the probability of $t > 4$ is almost 0.

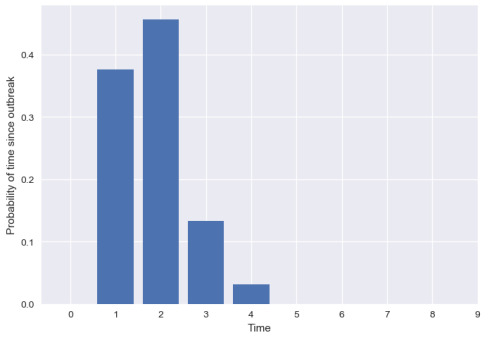

Figure 10: Probability distribution of time since infection, after the detection in Figure 8 is taken into account.

Finally, we estimate the marginals, a histogram of which can be seen in figure 11. This clearly shows that we have detected the infection in its early stage.

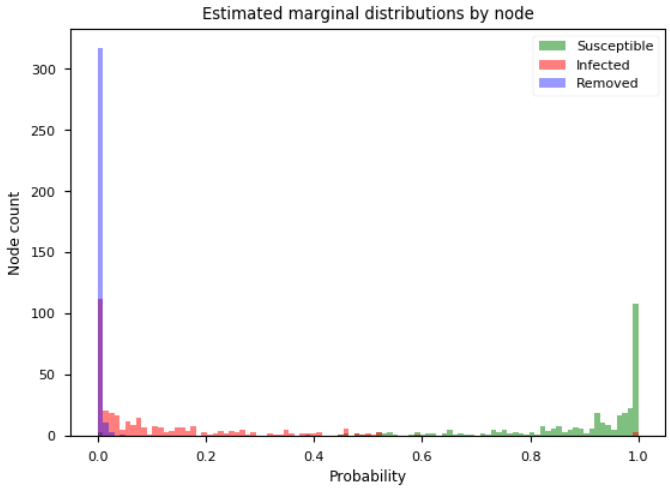

Figure 11: Histogram of the marginal distributions of each node, giving us a quick overview on the current status of the network after the detection in Figure 8. We see that most of the network is still likely healthy.

Finally, in Table 1 we can find the top nodes to test next based on entropy of the current estimated marginal distributions. As expected, the top airports are highly connected but non sensed airports

Airport	Susceptible probability	Infected probability	Removed probability	Entropy (bits)
Charlotte	0.447	0.504	0.049	1.230
Pittsburgh	0.478	0.500	0.022	1.131
Atlanta	0.456	0.522	0.022	1.128
Denver	0.460	0.522	0.018	1.108
Newark	0.500	0.487	0.013	1.088

Table 1: Top 5 highest entropy nodes after detection in Figure 8. If the tests are able to distinguish between all states, then these (in order) are the optimal nodes to test next if we want the maximum information gain about the network

6.2.2 Hypertext 2009

We perform a second experiment of another situation in a real human interaction dataset. We use the Hypertext 2009 network [18], a human interaction network. The ACM Conference on Hypertext and Hypermedia 2009 was held

in Turin, Italy in 2009, and during the conference, the participation badges included Radio-Frequency Identification devices embedded in conference badges that were able to mine face-to-face proximity relations [19]. The exchange of radio packets between badges implies a proximity of less than 1-1.5 m, a distance in which epidemics could spread. In the network, a node represents a conference visitor, and an edge represents a face-to-face contact that was active for at least 20 seconds. The network has $n = 113$ nodes and $m = 2196$ edges once we aggregate edges over time.

The flexibility of our framework allows us to also model a situation in which the sensors are not placed beforehand and we detect the epidemic late. The scenario here would be the following: a new epidemic arises, able to go unnoticed to general tests. We observe that someone dies at some point in time and the new infection is discovered as the cause of their death. What can a dataset of social interaction and tracing tell us about the state of the disease in the population? How likely is that the first person who died was the first person to contract the disease? What is the probability of infection of each node?

This small network plays the role of a dataset of social tracing that could be obtained nowadays with mobile phones, for example. The way to simulate this scenario in our model is to jump straight to the a posteriori questions, assuming that we have sensors placed in *all* nodes but they can only detect once someone is removed. Figure 12 shows the topology of the network, with the detected removed node marked in red.

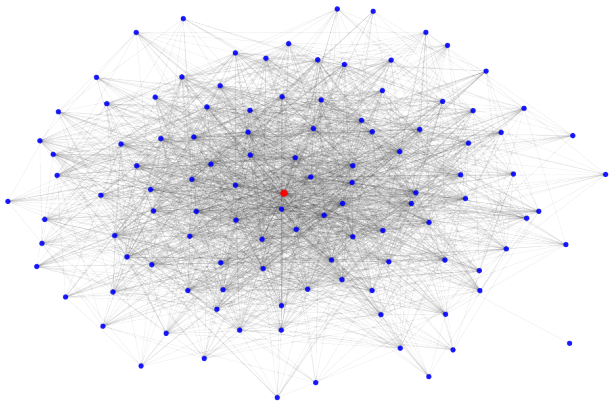

Figure 12: Topology of the Hypertext 2009 network, which has $n = 113$ nodes representing attendees to the conference and $m = 2196$ edges representing social interactions between attendees. The node that was first removed due the epidemic is marked in red

In figure 13, we see the results of the approximate inference of patient zero. We see that the probability of the removed node being patient zero is roughly 0.4, while the rest of nodes with positive probabilities have probabilities smaller than or around 0.05, with one exception. A highly disconnected node, appearing in the bottom right, is given an unusual high probability of around 0.2. This is because this node is *only* connected to the removed node, and so, if the infection actually started there, with very high probability itself or the true removed node would be the first deaths.

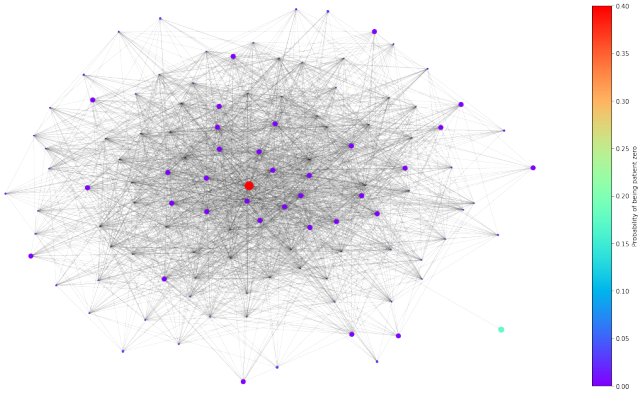

Figure 13: Approximate inference of patient 0 of the disease, with node color and size depending on the probability. The first removed node is patient zero with a probability of around 0.4

As for the time distribution, shown in Figure 14, the most relevant fact is that due to high δ it is unlikely that the infection has been going on for > 2 timesteps and that the 0 probability assigned to $t = 0$ comes in this case not because simultaneity but because the sensors sense the removed and not the infected state.

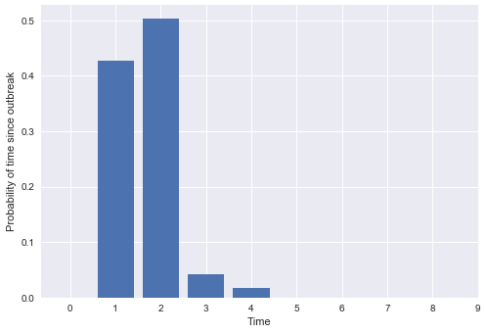

Figure 14: Posterior distribution of time since infection for the Hypertext2009 network

Finally, the most important question in this hypothetical situation would be the assessment of the current status of the network via marginal estimation. Due to high β and high connectivity of the removed node, we expect these probabilities to be quite high, and in fact, in figure 15 we see how clearly the probability depends on the graph distance from the removed node, with all the people that have socially interacted with the removed node having an infection probability around $0.6 - 0.7$ and the people which didn't interact directly with the removed node (all of which are at distance 2 to the removed node) have a far lower probability of around $0.2 - 0.3$. Comparing it with the histogram of the US air transportation dataset, we clearly see that in this case we have detected the infection much later, as the majority of the network is likely infected in this case.

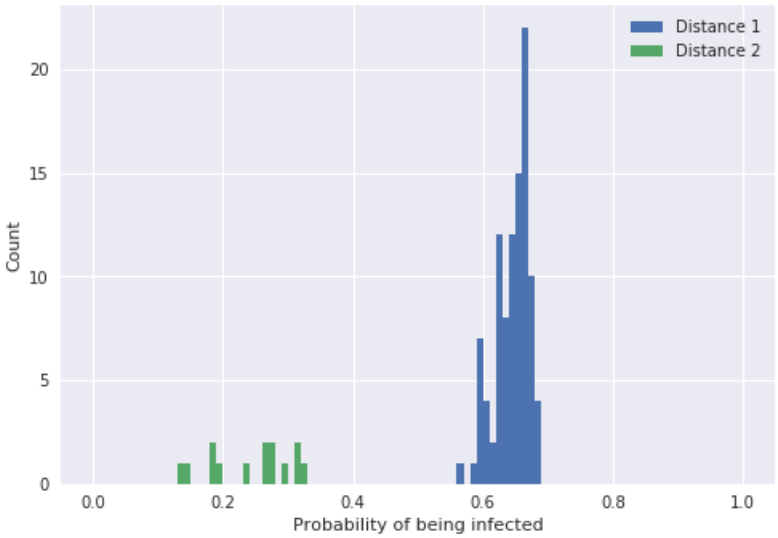

Figure 15: Histogram of infection probabilities separated by distance to the removed node

Similar as in the other example, we could calculate the entropies to decide further testing. Since we know that all testable nodes are not removed, the entropy is just a measure of how close the probability of infection is to 0.5. We can see from the graph that the first node to test would be the neighbor with lowest probability of infection.

7 Conclusions

In this work we have introduced a framework that uses the language of Markov Chains to solve problems in early detection of epidemics. Most importantly, the framework is model agnostic as long as the model can be expressed as a Markov Chain, which as commented is a weak assumption. Being able to model the sensors as well provides a huge flexibility and the same framework can be used to answer different questions. For the sensor placement problem, we have proved a submodularity result for a coverage-like function related to the cumulative density function of hitting times for any Markov Chain, which then can be used in this case as a guarantee that a greedy scheme achieves a constant bound approximation error with a quadratic number of function evaluations. After the sensor placement, we have used Bayesian inference algorithms to answer questions about the present status of the network and the origin of the epidemic. Finally, we propose a method to evaluate which tests to do next based on the estimation of the marginal distributions and the information gain of tests. The Monte Carlo algorithms proposed can then be used to scale to bigger graphs. We then have used the approximate algorithms in real networks to exemplify the type of insights that this framework can provide in real life scenarios, in one example in which we have previously prepared for an infection and one in which we have not.

References

[1] P. Van Mieghem. "Epidemic phase transition of the SIS type in networks". In: *EPL (Europhysics Letters)* 97.4 (Feb. 2012), p. 48004. DOI: 10.1209/0295-5075/97/48004. URL: https://doi.org/10.1209/0295-5075/97/48004.

[2] Yang Wang et al. "Epidemic spreading in real networks: an eigenvalue viewpoint". In: *22nd International Symposium on Reliable Distributed Systems, 2003. Proceedings.* IEEE Comput. Soc. DOI: 10.1109/reldis.2003.1238052. URL: https://doi.org/10.1109/reldis.2003.1238052.

[3] P. Van Mieghem, J. Omic, and R. Kooij. "Virus Spread in Networks". In: *IEEE/ACM Transactions on Networking* 17.1 (Feb. 2009), pp. 1–14. DOI: 10.1109/tnet.2008.925623. URL: https://doi.org/10.1109/tnet.2008.925623.

[4] Lauren Ancel Meyers. "Contact network epidemiology: Bond percolation applied to infectious disease prediction and control". In: *Bulletin of the American Mathematical Society* 44.01 (Oct. 2006), pp. 63–87. DOI: 10.1090/s0273-0979-06-01148-7. URL: https://doi.org/10.1090/s0273-0979-06-01148-7.

[5] M. E. J. Newman. *Exact Solutions of Epidemic Models on Networks*. Working Papers 01-12-073. Santa Fe Institute, Dec. 2001. URL: https://ideas.repec.org/p/wop/safiwp/01-12-073.html.

[6] István Z. Kiss, Joel C. Miller, and Péter L. Simon. *Mathematics of Epidemics on Networks*. Springer International Publishing, 2017. DOI: 10.1007/978-3-319-50806-1. URL: https://doi.org/10.1007/978-3-319-50806-1.

[7] Cameron Nowzari, Victor M. Preciado, and George J. Pappas. *Analysis and Control of Epidemics: A survey of spreading processes on complex networks*. 2015. eprint: arXiv:1505.00768.

[8] Tom Britton. "Epidemic models on social networks – with inference". In: *arXiv: Populations and Evolution* (2019).

[9] Devavrat Shah and Tauhid Zaman. "Rumors in a Network: Who's the Culprit?" In: *IEEE Transactions on Information Theory* 57.8 (Aug. 2011), pp. 5163–5181. DOI: 10.1109/tit.2011.2158885. URL: https://doi.org/10.1109/tit.2011.2158885.

[10] Jure Leskovec et al. "Cost-effective outbreak detection in networks". In: *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '07*. ACM Press, 2007. DOI: 10.1145/1281192.1281239. URL: https://doi.org/10.1145/1281192.1281239.

[11] Stephen P. Borgatti. "Centrality and network flow". In: *Social Networks* 27.1 (Jan. 2005), pp. 55–71. DOI: 10.1016/j.socnet.2004.11.008. URL: https://doi.org/10.1016/j.socnet.2004.11.008.

[12] Fern Y. Hunt. "Finding Optimal Sinks for Random Walkers in a Network". In: *CoRR* abs/1704.02365 (2017). arXiv: 1704.02365. URL: http://arxiv.org/abs/1704.02365.

[13] "A contribution to the mathematical theory of epidemics". In: *Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character* 115.772 (Aug. 1927), pp. 700–721. DOI: 10.1098/rspa.1927.0118. URL: https://doi.org/10.1098/rspa.1927.0118.

[14] Zhaoyang Zhang et al. "Modeling Epidemics Spreading on Social Contact Networks". In: *IEEE Transactions on Emerging Topics in Computing* 3.3 (Sept. 2015), pp. 410–419. DOI: 10.1109/tetc.2015.2398353. URL: https://doi.org/10.1109/tetc.2015.2398353.

[15] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher. "An analysis of approximations for maximizing submodular set functions—I". In: *Mathematical Programming* 14.1 (Dec. 1978), pp. 265–294. DOI: 10.1007/bf01588971. URL: https://doi.org/10.1007/bf01588971.

[16] L. A. Wolsey. "An analysis of the greedy algorithm for the submodular set covering problem". In: *Combinatorica* 2.4 (Dec. 1982), pp. 385–393. DOI: 10.1007/bf02579435. URL: https://doi.org/10.1007/bf02579435.

[17] Christian M. Schneider et al. "Suppressing Epidemics with a Limited Amount of Immunization Units". In: (2011). DOI: 10.1103/PhysRevE.84.061911. eprint: arXiv:1102.1929.

[18] *Hypertext 2009 network dataset – KONECT*. Sept. 2016. URL: http://konect.uni-koblenz.de/networks/sociopatterns-hypertext.

[19] Lorenzo Isella et al. "What's in a crowd? Analysis of face-to-face behavioral networks". In: *Journal of Theoretical Biology* 271.1 (Feb. 2011), pp. 166–180. DOI: 10.1016/j.jtbi.2010.11.033. URL: https://doi.org/10.1016/j.jtbi.2010.11.033.

A Proof of Theorem 1

Lemma 1. *Let S be a finite set and consider M a Markov Chain over the states of S. Then, for $\tau \in \mathbb{R}^+$ the function*

$$\begin{aligned} h\colon \mathcal{P}(S) &\to [0,1] \subset \mathbb{R} \\ W &\to \mathbb{P}\left(T_W \leq \tau\right) \end{aligned}$$

is submodular.

Note that this is different than f defined above in that the domain set is the power set of states and not the power set of vertices mapped to the power set of states via D.

Proof. We want to see that for $X \subset Y$ and $x \in S \setminus Y$

$$\mathbb{P}(T_{X\cup\{x\}} \leq \tau) - \mathbb{P}(T_X \leq \tau) \geq \mathbb{P}(T_{Y\cup\{x\}} \leq \tau) - \mathbb{P}(T_Y \leq \tau) \tag{30}$$

We define a path as a sequence of Markov Chain states $X_0, X_1, ...$ We only need to consider paths of length τ as all the events are independent of the state of the Markov Chain for $t > \tau$. We have

$$\begin{aligned} &\mathbb{P}(X_0 = x_0, \dots, X_\tau = x_\tau) = \\ &\mathbb{P}(X_0 = x_0) \prod_{i=1}^{\tau} \mathbb{P}(X_i = x_i | X_{i-1} = x_{i-1}) = \\ &\mathbb{P}(X_0 = x_0) \prod_{i=1}^{\tau} \mathbb{P}(X_1 = x_i | X_0 = x_{i-1}) \end{aligned} \tag{31}$$

The probability of the absorbing time to $Z \subset S$ being smaller or equal than τ is equal to the sum of probabilities of those paths that have an element of Z in their first τ elements after the initial state. This means that $\mathbb{P}(T_{X\cup\{x\}} \leq \tau) - \mathbb{P}(T_X \leq \tau)$ is the sum of probabilities of those paths that contain $\{x\}$ but do not contain any element in X, and likewise $\mathbb{P}(T_{Y\cup\{x\}} \leq \tau) - \mathbb{P}(T_Y \leq \tau)$ is the sum of probabilities of those paths that contain $\{x\}$ but do not contain any element in Y. Since $X \subset Y$ the second set is contained in the first, and so in the left hand side we are adding up the probabilities of at least the same paths than in the right hand side, and hence the inequality holds. □

The second part of the submodularity proof for f concerns being able to conserve the submodularity of h under composition with functions of certain properties.

Lemma 2. *(Conservation of submodularity under pullback). Let V, S be finite sets, and let $h\colon \mathcal{P}(S) \to \mathbb{R}$ be monotone submodular. Let $g\colon \mathcal{P}(V) \to \mathcal{P}(S)$ be a function satisfying $g(A \cup B) = g(A) \cup g(B)$ and $A \subset B \implies g(A) \subset g(B)$ for $A, B \subset V$. Then, $h \circ g\colon \mathcal{P}(V) \to \mathbb{R}$ is monotone submodular.*

(Note that by $g(A)$ we do not mean $\{g(x)|x \in A\}$ but rather the image of g of A as an element $g(\{A\})$, but we omit the brackets. A is a subset of V but an element of $\mathcal{P}(\mathcal{V})$, and therefore g sends it to a subset of S, element of $\mathcal{P}(S)$)

Figure 16 provides a scheme of the situation, in which we have used the explicit notation.

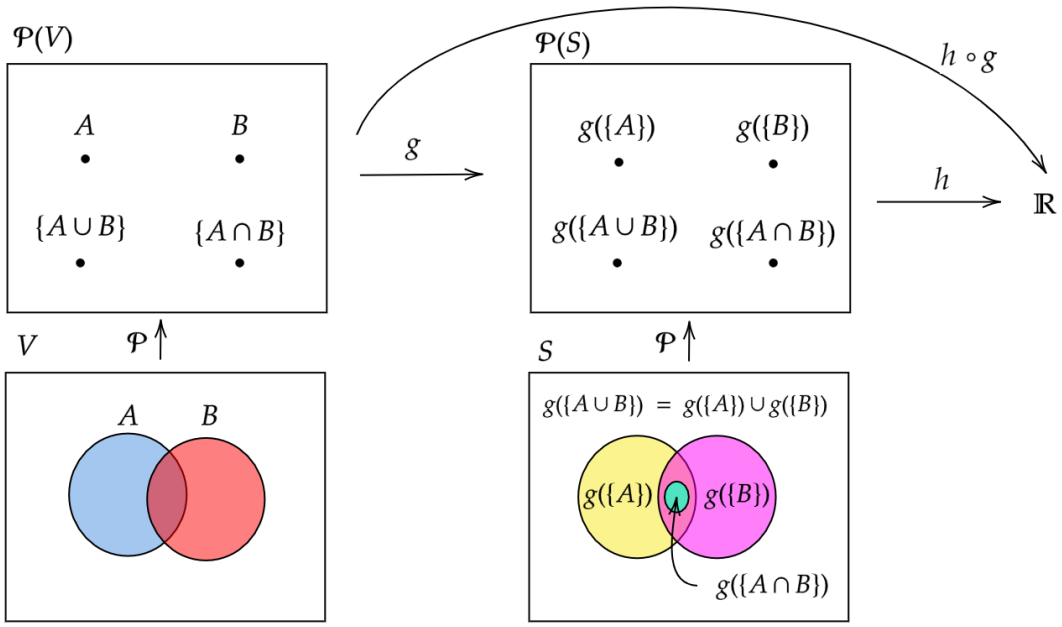

Figure 16: Situation of Lema 2. We want to see that given that h is monotone submodular and g satisfies certain conditions, the submodularity is conserved under the composition $h \circ g$

Proof. The monotonicity of $h \circ g$ comes from the monotonicity of h and g. For the submodularity, we want to see that for all $S, T \subset V$,

$$\begin{aligned}(h \circ g)(S) + (h \circ g)(T) &\geq (h \circ g)(S \cup T) + (h \circ g)(S \cap T) \\ h(g(S)) + h(g(T)) &\geq h(g(S \cup T)) + h(g(S \cap T)) \\ h(g(S)) + h(g(T)) &\geq h(g(S) \cup g(T))) + h(g(S \cap T))\end{aligned}$$

Since h is submodular and $g(S)$ and $g(T)$ are subsets of S, we know that

$$h(g(S)) + h(g(T)) \geq h(g(S) \cup g(T))) + h(g(S) \cap g(T))$$

But since g is monotonous we have $g(S \cap T) \subset (g(S) \cap g(T))$ and since h is monotonous $h(g(S \cap T)) \leq h(g(S) \cap g(T))$ and so we have $h(g(S)) + h(g(T)) \geq h(g(S) \cup g(T))) + h(g(S \cap T))$ □

Proof of Theorem 1: We apply Lemma 2 to the composition $h \circ D$, where D is the detection set function $D \colon \mathcal{P}(\mathcal{V}) \to \mathcal{P}(\mathcal{S})$ mapping W to D_W, where V is the set of vertices in the graph and S the set of Markov Chain states. By Lemma 1 h is submodular and it is also clearly monotonous by the same argument that we have shown that f is monotonous. By definition of the detection set function D, $D_A \subset D_B$ if $A \subset B$ and $D_{A \cup B} = D_A \cup D_B$. Therefore, $f = h \circ D$ is submodular. □.